

IN THE CLAIMS

Please cancel claims 1-40 and amend the claims as follows:

1-40. (Canceled)

41. (Currently amended) A method for creating a graphic user interface (GUI) for a computer application, comprising:

defining user interface objects and user interface elements corresponding to the user interface objects in a GUI layer that is separate from the computer application;

arbitrarily defining a mapping between the user interface objects and respective locations in a user interface screen, wherein the mapping ~~is independent~~ defines the locations independently of the computer application;

generating the user interface screen comprising the user interface elements in the respective locations determined by the mapping; and

linking the user interface objects in the GUI layer to methods in the computer application, so that interaction of a user with the user interface elements in the user interface screen invokes the methods linked to the objects.

42. (Previously presented) A method according to claim 41, wherein arbitrarily defining the mapping comprises:

defining a relation that associates the user interface objects with corresponding features in an image provided by a designer of the GUI;

receiving the image from the designer, wherein the image comprises one or more of the features in respective positions; and

determining the respective locations of the user interface objects based on the respective positions of the corresponding features in the image.

43. (Previously presented) A method according to claim 42, wherein defining the relation comprises associating the features of a given color with a corresponding one of the user interface objects.

44. (Previously presented) A method according to claim 43, wherein associating the features of the given color comprises identifying a certain color with a background region of the user interface screen, and wherein generating the user interface screen comprises displaying the background region as a transparent region.

45. (Previously presented) A method according to claim 42, wherein defining the relation comprises associating the features of a given shape with a corresponding one of the user interface objects.

46. (Currently amended) A method according to claim 42, wherein receiving the image comprises changing a graphic quality of one of the features in the image, and
wherein generating the user interface screen comprises changing the corresponding user interface element on the user interface screen responsive to changing the graphic quality, ~~substantially~~ without effect on the method to which the corresponding user interface object is linked.

47. (Previously presented) A method according to claim 46, wherein changing the graphic quality comprises changing the position of the one of the features in the image, and

wherein changing the corresponding user interface element comprises changing the location of the user interface element in the user interface screen.

48. (Previously presented) A method according to claim 46, wherein changing the graphic quality comprises changing a size characteristic of the one of the features in the image, and

wherein changing the corresponding user interface element comprises changing a corresponding size characteristic of the user interface element in the user interface screen.

49. (Previously presented) A method according to claim 42, wherein receiving the image comprises receiving a bitmap image, such that the respective locations of the user interface objects are determined responsive to the bitmap image.

50. (Currently amended) A method according to claim 42, wherein generating the user interface screen comprises building the user interface screen based on the features of the image received from the designer, ~~substantially~~ without resort to a textual description of the user interface elements.

51. (Previously presented) A method according to claim 42, wherein defining the relation comprises identifying at least one of the features in the image with a user interface push button.

52. (Previously presented) A method according to claim 42, wherein defining the relation comprises identifying at least one of the features in the image with an area for display of text or graphics associated with the application.

53. (Previously presented) A method according to claim 42, wherein defining the relation comprises identifying at least one of the features in the image with a user control for selecting a value of a parameter from a range of values.

54. (Previously presented) A method according to claim 53, wherein the at least one of the features in the image comprises a range of colors corresponding to the range of values of the parameter.

55. (Previously presented) A method according to claim 53, wherein the at least one of the features in the image defines a range of positions of a slider corresponding to the range of values of the parameter.

56. (Currently amended) A method according to claim 55, wherein the at least one of the features in the image comprises an elongate feature that deviates ~~substantially~~ from a straight, linear shape.

57. (Previously presented) A method according to claim 41, wherein defining the user interface objects comprises altering one of the user interface objects by inheritance thereof.

58. (Previously presented) A method according to claim 41, wherein generating the user interface screen comprises providing a skin including graphic representations of the user interface elements at the locations to which the corresponding user interface objects are mapped.

59. (Previously presented) A method according to claim 41, wherein generating the user interface screen comprises altering an appearance of one or more of the user interface elements while the application is running.

60. (Previously presented) A method according to claim 59, wherein altering the appearance comprises providing multiple different views of the user interface screen.

61. (Previously presented) A method according to claim 60, wherein providing the multiple different views comprises providing zoom-in and zoom-out views.

62. (Previously presented) A method according to claim 60, wherein the application comprises a multimedia player application having multiple channels, and wherein providing the multiple different views comprises associating the different views with different channels of the multimedia player.

63. (Previously presented) A method according to claim 59, wherein the application presents content to the user, and wherein altering the appearance comprises altering the appearance of the one or more user interface elements responsive to a characteristic of the content.

64. (Currently amended) A method according to claim 41, wherein arbitrarily defining the mapping comprises mapping the user interface objects in a manner that is ~~substantially~~ independent of an operating platform on which the application runs.

65. (Previously presented) A method according to claim 41, wherein generating the user interface screen comprises generating a browser screen on a computer accessing the application remotely via a network.

66. (Currently amended) A method according to claim 65, wherein generating the user interface screen comprises generating ~~substantially~~ the same user interface screen on the browser and on a local client of the application.

67. (Previously presented) A method according to claim 41, wherein arbitrarily defining the mapping comprises defining a relation that is preserved across multiple, different applications.

68. (Currently amended) A computer software product for creating a graphic user interface (GUI) for a computer application, the product comprising a computer-readable medium in which program instructions are stored, which instructions, when read by a computer, cause the computer to receive a definition of user interface objects and user interface elements corresponding to the user interface objects in a GUI layer that is separate from the computer application, and to receive an arbitrary definition of a mapping between the user interface objects and respective locations in a user interface screen, wherein the mapping ~~is independent~~ defines the locations independently of the computer application, the instructions further causing the computer to generate the user interface screen comprising the user interface elements in the respective locations determined by the mapping, and to link the user interface objects in the GUI layer to methods in the computer application, so that interaction of a user with the user interface elements in the user interface screen invokes the methods linked to the objects.

69. (Previously presented) A product according to claim 68, wherein the mapping is defined in terms of a relation that associates the user interface objects with corresponding features in an image provided by a designer of the GUI, and

wherein the instructions cause the computer to receive the image from the designer, wherein the image comprises one or more of the features in respective positions, and to determine the respective locations of the user interface objects based on the respective positions of the corresponding features in the image.

70. (Previously presented) A product according to claim 69, wherein the instructions cause the computer to associate the features of a given color with a corresponding one of the user interface objects.

71. (Previously presented) A product according to claim 69, wherein the instructions cause the computer to associate the features of a given shape with a corresponding one of the user interface objects.

72. (Currently amended) A product according to claim 69, wherein the instructions cause the computer, responsive to a change made by a user in a graphic quality of one of the features in the image, to change the corresponding user interface element on the screen responsive to changing the graphic quality, ~~substantially~~ without effect on the method to which the corresponding user interface object is linked.

73. (Currently amended) A product according to claim 68, wherein the instructions cause the computer to generate the user interface screen ~~substantially~~ without dependence on an operating platform on which the application runs.

74. (Previously presented) A product according to claim 68, wherein the mapping between the user interface objects and the respective locations in the user interface screen is preserved across multiple, different applications.

75. (Previously presented) A product according to claim 68, wherein the instructions enable the computer to alter an appearance of one or more of the user interface elements while the application is running.

76. (Currently amended) Apparatus for creating a graphic user interface (GUI) for a computer application, comprising:

a GUI processor, which is adapted to receive a definition of user interface objects and user interface elements corresponding to the user interface objects in a GUI layer that is separate from the computer application, and to receive an arbitrary definition of a mapping between the user interface objects and respective locations in a user interface screen, wherein the mapping ~~is independent~~ defines the locations independently of the computer application,

wherein the GUI processor is further adapted to generate the user interface screen comprising the user interface elements in the respective locations determined by the mapping, and to link the user interface objects in the GUI layer to methods in the computer application, so that interaction of a user with the user interface elements in the user interface screen invokes the methods linked to the objects; and

a display, which is driven by the processor to display the user interface screen.

77. (Previously presented) Apparatus according to claim 76, wherein the mapping is defined in terms of a relation that associates the user interface objects with corresponding features in an image provided by a designer of the GUI, and

wherein the processor is adapted to receive the image from the designer, wherein the image comprises one or more of the features in respective positions, and to determine the respective locations of the user interface objects based on the respective positions of the corresponding features in the image.

78. (Previously presented) Apparatus according to claim 77, wherein the processor is adapted to associate the features of a given color with a corresponding one of the user interface objects.

79. (Previously presented) Apparatus according to claim 77, wherein the processor is adapted to associate the features of a given shape with a corresponding one of the user interface objects.

80. (Currently amended) Apparatus according to claim 77, and comprising a pointing device, which is operable by the designer to change a graphic quality of one of the features in the image,

wherein the processor is adapted to change the corresponding user interface element on the screen responsive to changing the graphic quality, ~~substantially~~ without effect on the method to which the corresponding user interface object is linked.

81. (Currently amended) Apparatus according to claim 76, wherein the mapping between the user interface objects and the respective locations in the user interface screen is preserved across multiple, different applications, ~~substantially~~ without dependence on an operating system run by the processor.

82. (Previously presented) Apparatus according to claim 76, wherein the processor is adapted to alter an appearance of one or more of the user interface elements while the application is running.